

Tool Support for the Quality Assessment of MDWE Methodologies

F.J. Domínguez-Mayo, M.J. Escalona, M. Mejías, and J. Ponce

1 Introduction

An interesting concern in development teams of Web applications is how to build Web applications automatically, with the least cost and the best quality. In this line, the model-driven engineering (MDE) approach to software development focuses on creating models or abstractions closer to some particular domain concepts rather than computing concepts. It is meant to increase productivity by maximizing compatibility among systems, simplifying the design process and promoting communication between individuals and teams working on the system.

The rise of this paradigm has already influenced current Web developments. The most important research initiative in this area is the Model Driven Architecture (MDA) ([OMG: MDA](#)), which is being developed under the umbrella of the Object Management Group (OMG). According to this group, models may be independent from the characteristics of any technological platform. Besides, the life cycle of a software system is completely covered, starting from requirements capture, passing through code generation up to the system maintenance. In this context, Web engineering is a specific domain where model-driven software development can be successfully applied (Escalona and Aragón 2008). The use of MDE in Web engineering is called model-driven Web engineering (MDWE), and as it is noticed in different papers ([ISO- International Organization for Standardization, ISO/IEC 9126-1](#)), in the last years, several research groups have proposed different methodologies with processes, models and techniques to build applications such as UWE ([UML-Based Web Engineering](#)), WebML ([The Web Modelling Language](#)), OOHDM ([Oriented Hypermedia Design Method](#)), OOH4RIA (Meliá et al. 2010), Rux-Method (Preciado et al. 2008) or NDT ([Navigational Development Techniques](#)). Some of them involve most of the levels of abstraction, and they

even have tools that support the automation of transformations in the development processes. A suitable design is needed as well as updated MDWE methodologies and effective tools. To this end, our work focuses on evaluating and comparing existing proposals, bearing in mind that the framework could be extended to other areas in the future. In order to offer a suitable tool support for evaluating the quality of approaches, this chapter proposes a tool support for QuEF (quality evaluation framework) which is a framework for the quality evaluation of MDWE methodologies.

This chapter is organized into the following sections. Following this introduction, Sect. 2 offers an overview of QuEF. Section 3 describes the tool support description and basic proposed interfaces together with the users' context. Finally, Sect. 4 states a set of conclusions and contributions and proposes possible future work.

2 An Overview of QuEF

QuEF (Domínguez-Mayo et al. 2010a, b, c) is a framework for the assessment of MDWE methodologies. This approach is oriented towards the quality evaluation of MDWE methodologies in a specific environment by means of objective measures. Given the high number of methodologies available and proposed in recent years, it has become necessary to define objective evaluation tools which enable organizations to improve their methodological environment and help Web methodology designers to create new effective and efficient tools, processes and techniques. Due to the progressive evolution of methodologies, it is not only necessary to evaluate quality but also to find out how to improve and optimize this process in order to reduce costs. In this work, an approach or methodology for the development of Web applications is a model-driven proposal. It provides a set of guidelines, techniques, processes and/or tools for structuring specifications which are expressed as models.

There are only considered Web modelling approaches based on MDA in the framework. In addition, a framework is a basic conceptual structure composed of a set of elements used to evaluate, in this case, MDWE methodologies, although it could be extended to other area or domain. It consists of the following elements:

- *Quality Model*. It includes the basis for quality requirements specifications with the purpose of evaluating quality. It specifies each element and its purposes.
- *Analysis Process*. It includes the description templates for the evaluation of methodologies and defines the importance of each element. It depends on the Quality Model description.
- *Evaluation Process*. It includes the definition and specification to carry out the quality evaluation process.
- *Multi-Criteria Group Decision Making*. It includes the necessary elements to better the standardization of the access channel and communication among users

be met by means of the evaluation. Quality Characteristics and Quality Sub-Characteristics define the quality aspects of the required model. Finally, these Information Needs have to be mapped with the quality aspects.

- *Quality Characteristic*. This is a higher-level quality aspect. For instance, Usability, Maintainability, Functionality, Reliability or Portability could be considered Quality Characteristics.
- *Quality Sub-Characteristic*. In a hierarchy of quality aspects, this is a lower-level quality aspect. For instance, Usability is defined through several Quality Sub-Characteristics, such as Learnability, Understandability and Operability.

Each characteristic in ISO/IEC 9126 ([ISO- International Organization for Standardization, ISO/IEC 9126-1](#)) and ISO/IEC 25000 ([SQaRE](#)) is described in relation to a software product. However, in our particular case, all Quality Characteristics and Quality Sub-Characteristics are described regarding approach characteristics.

In MDWE, models are progressively refined and transformed into new models or codes. To this end, tools may also be useful to test, verify or validate models. Moreover, each methodology may define its development process and/or techniques. The quality of methodologies subsequently depends on the diverse Features, such as the *MDE*, the *Web Modelling*, the *maturity* of a methodology and the *tool support* applied to discover faults or weaknesses. The main idea is to characterize the whole MDWE process. The elements describing the Information Needs in the Quality Model are:

- *Feature*. This is a higher-level description concept of an approach. It may be, for example, the software development process, models, metamodels, languages, tools or the transformations used.
- *Sub-Feature*. This is a lower-level description concept of an approach. For example, the model-driven engineering is a Feature with several Sub-Features, such as the Language Definition, Transformations and Trace.
- *Metric*. In the Quality Model, metrics should measure the degree to which a Sub-Feature reaches the required model. To simplify, a metric is used for measuring Sub-Features.

The Analysis Process component would include the templates which have to be generated in terms of Quality Model description, specifically the Information Needs description. The methodology can be analysed with these templates considering the Weight value definition of Sub-Features and Quality Sub-Characteristics and the Matrix of Influences (MoI), which indicates the influence in Sub-Feature and Quality Sub-Characteristics. Finally, the Evaluation Process component performs the quality evaluation, and the Multi-Criteria Group Decision Making is responsible for reaching a consensus on the terminology and weight values among users.

Different basic interfaces are proposed and given in this chapter.

As far as user context is concerned, this tool support would have three different user contexts for QuEF-TS. As it is shown in [Fig. 22.2](#), the first one would be the Developer user, who cannot modify anything. It would have a general view of the

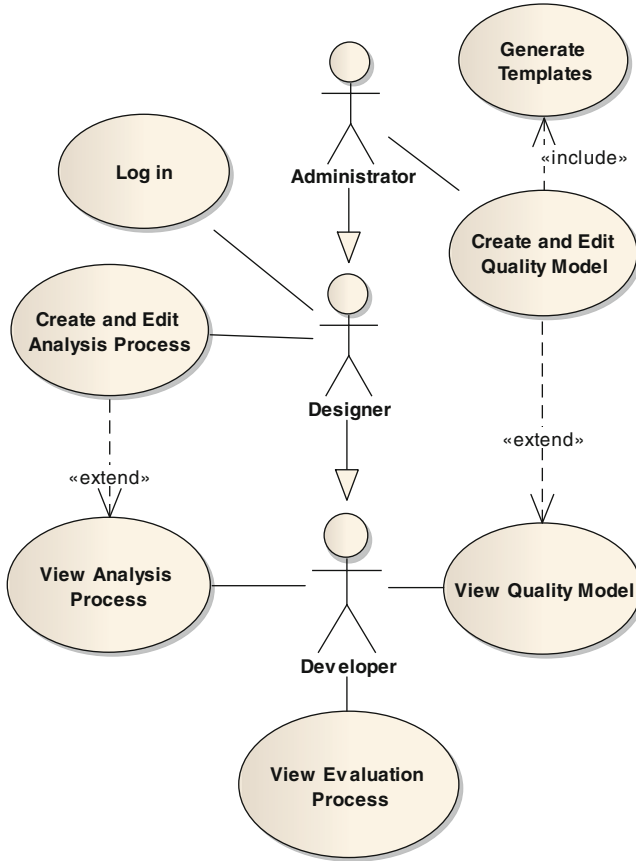


Fig. 22.2 Use cases for the proposed tool support

Quality Model, the Analysis Process and the Evaluation Process. The second one would be the Designer user, who would have more permission and could not only view but also create and edit the Analysis Process components such as the MoI, the Weight values for Quality Sub-Characteristics and Sub-Features and Template values. This user could also log in the tool support. Thus, it would save different MoIs, Weight values and Templates and exchange them with other Designers. It would be useful to evaluate methodologies according to other designers' points of view. Finally, the third user would be the Administrator who would create and edit the Quality Model and generate the templates based on it in order to be used in the Analysis Process.

QuEF-TS can generate templates and MoIs in terms of the Quality Model specified. Thus, an important issue is to define the Quality Model and its knowledge.

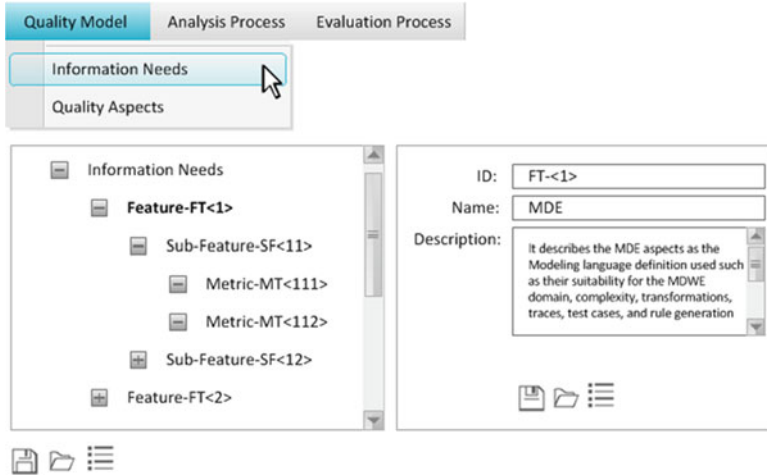


Fig. 22.3 Basic interface for information needs description

3.1 The Quality Model

Figure 22.3 represents the basic interface of Information Needs. This interface shows how the Features, Sub-Features and Metrics are defined. In order to build the different elements, a tree structure is proposed and used by the Administrator with the aim of creating and editing Features, Sub-Features and Metrics in terms of the Information Needs established by the community of designers. The definition of this hierarchy is currently being developed, although we are also working on Multi-Criteria Group Decision Making (MCGDM) (Preciado et al. 2008) to reach a consensual definition of the hierarchy.

A set of Quality Characteristics and Quality Sub-Characteristics for each Quality Characteristic based on current literature, such as ISO/IEC 9126 or other standards which are adapted to MDWE methodologies, are identified, classified and placed in a hierarchy. Likewise, the Quality Characteristics and Quality Sub-Characteristics are defined in the Quality Model, as shown in Fig. 22.4. On its left side, this figure shows a tree structure where the user would create and edit Quality Characteristics and Quality Sub-Characteristics in terms of the Quality Aspects considered. Furthermore, in order to define a Quality Model, it must contain association links between the Sub-Features and the Quality Sub-Characteristics. These links represent the dependencies between Sub-Features and Quality Sub-Characteristics and show how each Sub-Feature affects every Quality Sub-Characteristic. Association links may be based on proven real-world experience or determined by real case study applications in a number of real projects. We are currently working on a MCDGM method which could permit reaching a consensus value according to different experts' views.

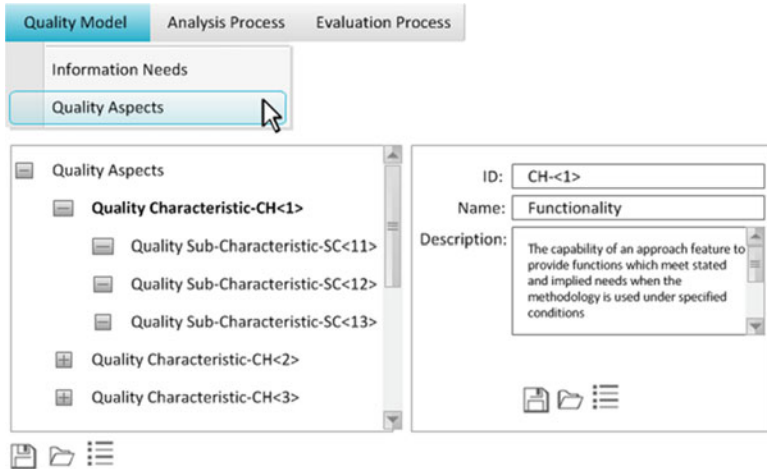


Fig. 22.4 Basic interface for quality aspects description

3.2 The Analysis Process

In this component, the association links between Sub-Features and Quality Sub-Characteristics are defined in the MoI, which indicates, by means of a figure or value, how every Sub-Feature affects Quality Sub-Characteristic.

For instance, Usability is described as a set of Quality Sub-Characteristics. As shown is Fig. 22.5, these Quality Sub-Characteristics could be affected by one of the Sub-Features. This figure also represents the basic interface for the MoI. Each value indicates the influences between Sub-Features and Quality Sub-Characteristics appearing in this figure for every Sub-Features of MDE, such as Levels of Abstraction, Standard Definition, Model-Based Testing, Transformations and Traces, as well as for Quality Sub-Characteristics of Usability such as Learnability, Understandability, Simplicity and Interpretability.

The templates in the *Analysis Process* are based on the Quality Model definition, and they are used to describe an input methodology. Figure 22.6 represents the suggested basic interface. These templates would be used as input in the environment, analysed in the evaluation process and compared with the required quality model of the Quality Model definition. Templates for MDE, Web Modelling, Tool Support and Maturity have already been developed. Finally, the *Evaluation Process* would contrast the information from each input approach template with the information from the Quality Model definition. The main evaluation purpose is to identify trade-offs and sensitivity points of the methodology under study. The aim is to determine which aspect needs to be improved on a MDWE methodology.

Each Sub-Feature has to define a set of quantitative or qualitative metrics used to measure each Sub-Feature. In these lines, it might be interesting to establish standard metrics on MDWE to centralize them. In the literature, numerous



Fig. 22.5 Basic interface of the matrix of influences (*MoI*)

references to metrics can be found, but standardization has not been carried out yet. The types of metric defined are either qualitative or quantitative. Qualitative metric range indicates if the metric is Supported, Partly Supported or Not Supported, as appeared in Fig. 22.6.

On the contrary, Fig. 22.7 offers the Weight values setting for the Information Needs and Quality Aspects. Weight value represents the importance of each Sub-Feature in the set of Sub-Features for a Feature. Similarly, a weight value is also used to define the importance of each Quality Sub-Characteristic in the set of Quality Sub-Characteristics for a Quality Characteristic. MGDCM methods are being currently developed to define weight values for Sub-Features and Quality Sub-Characteristics.

3.3 The Evaluation Process

The Evaluation Process is the component which users can evaluate whether they have reached the required model of quality or not. The proposed basic interface, as shown in this example of Fig. 22.8, considers the Sub-Features of Web Modelling.

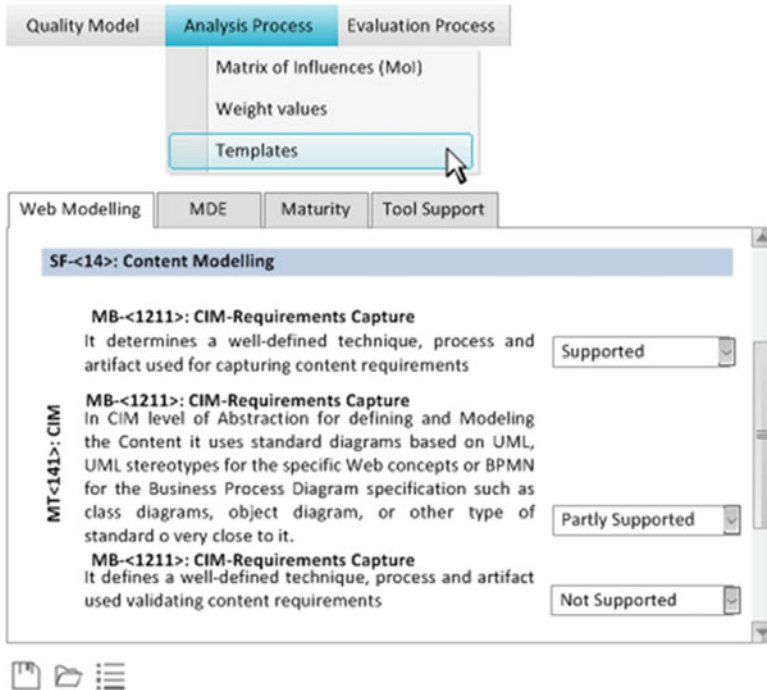


Fig. 22.6 Basic interface of the templates

We can see the Sub-Feature values for the Web Conceptual Levels, Interfaces, Development Process, Content Modelling, Presentation Modelling, Navigation Modelling and Business Modelling. All of them are Web Modelling Sub-Features. In other words, the views of all Sub-Feature values of a specific Feature or a view of all Feature values in general can be shown with this interface.

Quality Characteristics and Quality Sub-Characteristics are shown in the same way that Features and Sub-Features appeared, although in this interface, the user has to indicate which Features has to be considered in order to calculate the Quality Sub-Characteristics and Quality Characteristics.

The reason is that these values depend on the influences that Sub-Features have on Quality Sub-Characteristics. In other words, the value definition of Quality Characteristics and Quality Sub-Characteristics is determined by the relationship (s) between the Information Needs and the Quality Aspects. Figure 22.9 represents the basic interface for Quality Aspects and provides a general view of Quality Characteristics. We can also obtain all Quality Sub-Characteristic values for a Quality Characteristic by selecting the appropriate Data value option. In this example, we can observe all Quality Characteristic values in terms of Web Modelling, MDE and Maturity, which are the Features that have been selected in “In terms of” option.

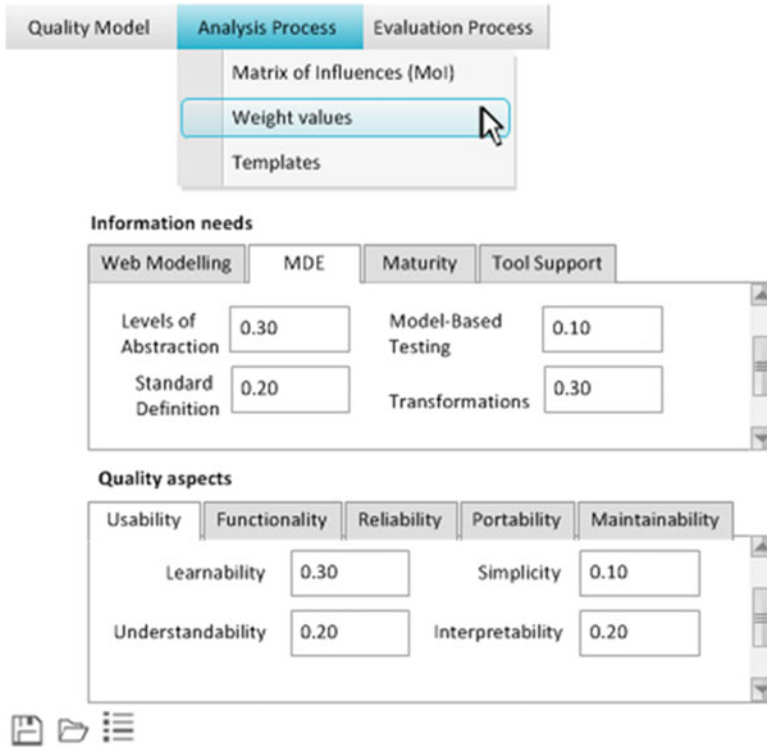


Fig. 22.7 Basic interface of the weight values setting

4 Conclusion

A tool support which implements the QuEF framework is needed for the improvement of current proposals and would be highly useful for the successful development of a new MDWE methodology. In this chapter, QuEF-TS, a proposed tool support to implement QuEF, is presented, and different basic interfaces are described. We consider that using this tool support will enhance the quality of products, processes and techniques of approaches because designers can analyse, control and improve the quality of their approaches easily. This tool can be used for the cost optimization in the improvement of methodologies since the Sub-Features selected to be implemented can be reduced by means of the Matrix of Influences (MoI) because we would take into account the minor number of Sub-Features with the major influence in Quality Sub-Characteristics. Thus, the use of QuEF-TS could improve the efficiency and effectiveness of MDWE methodologies and, consequently, may lead to a more widespread use due to the fact that this evaluation approach helps people to understand the strengths and weaknesses of a methodology. We consider it necessary to carry out a standardization of terminology to improve the access channel for communication on MDWE. In this context, we are

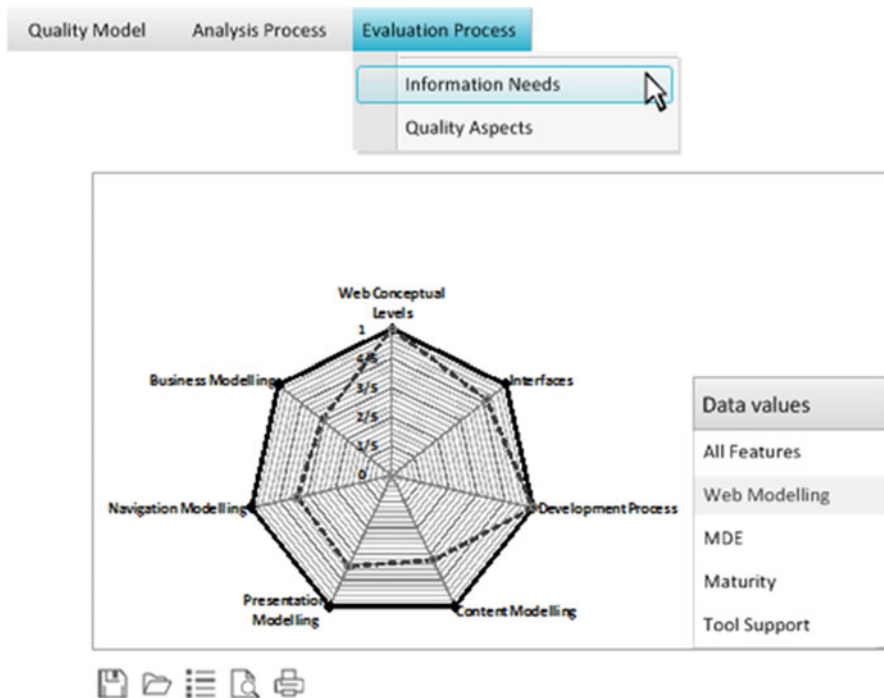


Fig. 22.8 Basic interface of information needs for the evaluation process

currently working in MCGDM methods to reach a consensus on methodology designers groups, as well as on the definitions of several aspects, such as Quality Model (Feature, Sub-Features, Metrics, Quality Characteristics and Quality Sub-Characteristics), Weight values and the influence of each Sub-Feature on each Quality Sub-Characteristics.

Other future work will be the integration of QuEF-TS in well-known social networks like Facebook ([Facebook for developers](#)) or Twitter ([Twitter for developers](#)), to make easier the MCGDM methods and reach this consensus more easily among different users.

Finally, another important aspect is that this environment could involve different areas or domains since QuEF-TS can generate the templates and all necessary elements based on the Quality Model description in order to analyse, control and evaluate the specified domain.

Acknowledgements This research has been supported by the project QSimTest (TIN2007-67843-C06_03) and by the Tempros project of the Ministerio de Educación y Ciencia, (TIN2010-20057-C03-02), Spain.

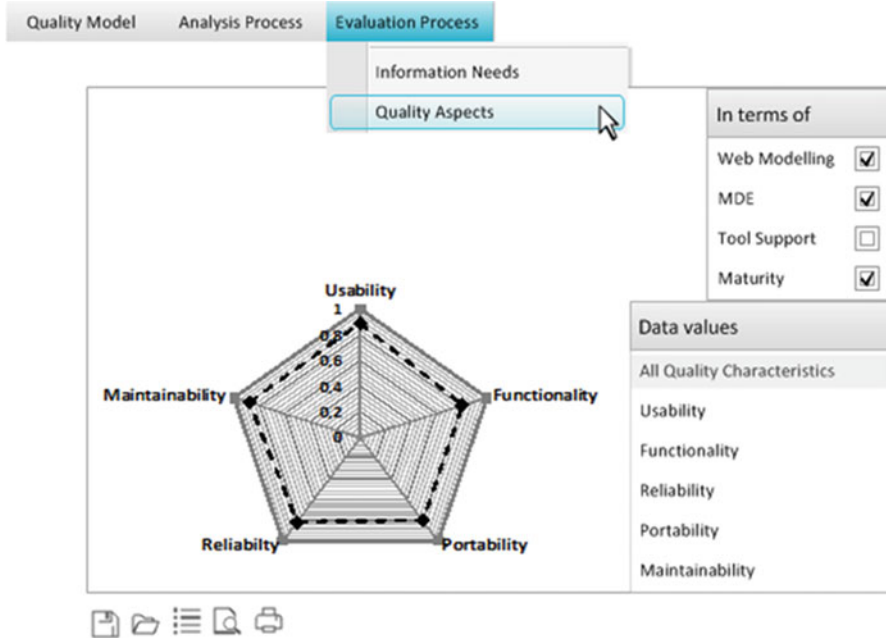


Fig. 22.9 Basic interface of the quality aspects for the evaluation process

References

- Domínguez-Mayo FJ, Escalona MJ, Mejías M (2010a) QuEF (quality evaluation framework) for model-driven web methodologies. In: Proceeding ICWE'10, Proceedings of the 10th international conference on current trends in web engineering, pp 571–575. ISBN:3-642-16984-8 978-3-642-16984-7
- Domínguez-Mayo FJ, Escalona MJ, Mejías M, Ramos I, Fernández L (2010b) A quality evaluation framework for MDWE methodologies. In: Proceedings of the 18th international conference on software quality management (SQM 2010), vol 1, No. 1, pp 171–184
- Domínguez-Mayo FJ, Escalona MJ, Mejías M, Torres AH (2010c) Towards a quality evaluation framework for model-driven web engineering methodologies. In: Sixth international conference on web information systems and technologies (WEBIST 2010), vol 2, pp 191–194
- Escalona MJ, Aragón G (2008) NDT. A model-driven approach for web requirements. *IEEE Trans Softw Eng* 34(3):377–390
- Facebook for developers. Retrieved April 2011, from <http://developers.facebook.com>
- ISO- International Organization for Standardization, ISO/IEC 9126-1. <http://www.iso.org>. Retrieved January 2010, from <http://www.iso.org>
- ISO/IEC 25000:2005. Software engineering – software product quality requirements and evaluation (SQuaRE) – guide to SQuaRE
- Meliá S, Pardillo J, Cachero C (2010) Automatic selection of RIA software architectures using quality models. *Quality of information and communications technology (QUATIC)*, 2010. In: Seventh international conference on the quality of information and communications technology (QUATIC), pp 505–510. ISBN: 978-1-4244-8539-0
- NDT (Navigational Development Techniques). Retrieved March 2011, from <http://www.iwt2.org/en/ndt.php>

OMG: MDA. Retrieved January 2010, from <http://www.omg.org/mda/>

OOHDM (Oriented Hypermedia Design Method). Retrieved March 2011, from <http://www-di.inf.puc-rio.br/schwabe/HT96WWW/section1.html>

Preciado JC, Linaje M, Morales-Chaparro R, Sánchez-Figueroa F, Zhang G, Kroiß C, Koch N (2008) Designing rich internet applications combining UWE and RUX-Method. In: ICWE '08 Proceedings of the 2008, 8th international conference on web engineering. ISBN:978-0-7695-3261-5

Saaty TL (1983) Introduction to a modeling of social decision process. Math Comput Simul 25:105–107

Twitter for developers. Retrieved April 2011, from <http://dev.twitter.com>

UWE (UML-Based Web Engineering). Retrieved March 2011, from <http://uwe.pst.ifi.lmu.de>

WebML (The Web Modelling Language). Retrieved March 2011, from <http://www.Webml.org>